Revue d'Intelligence Artificielle
Vol. 38, No. 3, June, 2024, pp. 1017-1026
Journal homepage: http://iieta.org/journals/ria

An Influence Maximization in Social Networks Based on Community Structure (IMSC)

Srinu Dharavath[1*], Natarajasivan Devarajan[1], Nalini Chidambaram[2]

[1] Department of Computer Science and Engineering, Annamalai University, Annamalai Nagar 608002, India
[2] Department of Computer Science and Engineering, BIHER, Chennai 600073, India

Corresponding Author Email: srinudharavathresearchscholar@gmail.com

https://doi.org/10.18280/ria.380327

Received: 11 September 2023
Revised:11 January 2024
Accepted: 3 February 2024
Available online: 21 June 2024

Keywords:
influence maximization, influence maximization in social networks

ABSTRACT

Recognition of the top k users (seed) in social networks is a challenge in influence maximization (IM), which aims to increase influence propagation. Finding the network's most important nodes can help network analysts track the spread of rumors, diseases, and data in many applications. We provide the Influence Maximization in Social Networks Based on Community Structure (IMSC) technique to address this problem. The three phases of our suggested framework, IMSC, are as follows: the network's community structure is identified, candidates are generated using community information, and seed nodes are finally selected from the candidate set. In this phase, the IMSC method looks at a network or group of interconnected things (like social connections or data points). It determines how they naturally form smaller communities or groups. After identifying these groups, the method generates a list of potential candidates or members who could play a key role in spreading data or influence within each community. Finally, from the list of potential candidates, the method selects a few critical individuals called "seed nodes." These seed nodes are starting points for spreading information or influence throughout the network. Our tests on real-world datasets display that the proposed method surpasses the competition in terms of the value of the corresponding output influential nodes while still using an acceptable amount of memory and processing time for massive graphs. We assess our algorithm against state-of-the-art solutions to the influence maximization issue.

1. INTRODUCTION

By adding billions of devoted users, social networks have developed into potent platforms for disseminating knowledge and marketing. The social impact, which charts the interactions between people in the network and can be assessed based on reputation and trust, is an underlying factor supporting the capabilities [1]. Marketing, which recognizes the significant impact of "word-of-mouth" that develops the interpersonal influence of relationships with customers and can change consumers' attitudes and behaviors, is a common application that social networks encourage [2, 3].

The proliferation of internet communication channels has facilitated the quick and straightforward transmission of emotions, information, ideas, and other experiences, which has given influence over decisions to buy (or adopt) products a significant amount of weight [4]. Even though social influence has always been acknowledged as a component in decision-making, contact can now be easily traced because of conveniently accessible internet customer footprints. Social networks are essential for transmitting knowledge, influence, and opinions among their users. As a result, there is a lot of interest in comprehending the dynamics of adoption within a social network because it could provide information on more effective marketing tactics.

Through word-of-mouth, the social network is exposed to influence or information [5]. A significant problem in analyzing and researching the social network is information dissemination. Based on this topic, the influence maximization (IM) problem is created. With IM, the issue is reaching a select group of power users who can disseminate their influence throughout the network the fastest [6, 7]. Numerous uses for choosing influential users exist, including network monitoring, social recommendation, rumor management, viral marketing, and income maximization.

Promoting and selling their products are essential to the survival of many commercial enterprises. For this reason, they are always looking for new strategies to advertise their products efficiently. This goal might be attained by word-of-mouth and the appropriate context-based dissemination of advertising messages by carefully chosen individuals [8]. Online social networks are suitable options for disseminating product marketing. These networks' captivating surroundings have drawn many users, multiplying daily. The characteristics of online social networks, where a seed set of users is chosen to start distributing the content on the network, can significantly assist viral marketing [9, 10]. There are only a few seed members because businesses have restricted advertising expenditures. Here, the issue of efficiently finding the most influential users and adding them to the seed set arises. A group of k users on which the spreading process is started must be identified to solve the IM issue, which is what this issue is known as. Although user behavior and interests have been considered essential components in the spreading process

in various techniques, such information is only sometimes present in actual networks. Therefore, to identify prominent users, we frequently only have network structural information at our disposal [11, 12].

The Influence Maximization with Monarch Butterfly Optimization Algorithm (IMSC) is a novel strategy aimed at enhancing the identification of influential nodes within communities. Building on the foundations laid by recent advancements in network analysis, IMSC offers a promising solution to the challenges posed by existing methods. Community structure identification, a crucial initial step, has often been approached with conventional algorithms. IMSC, however, employs the Monarch Butterfly Optimization Algorithm to delineate community boundaries meticulously. This ensures a more accurate representation of social clusters and serves as a departure from the limitations associated with some widely used techniques [13]. Furthermore, IMSC's candidate generation phase optimizes the selection of potential influencers within identified communities. Unlike traditional methods that may rely on heuristic approaches, IMSC's use of the Monarch Butterfly Optimization Algorithm enables a more nuanced network exploration, systematically identifying individuals with a higher potential for influence. Seed node selection, the ultimate determinant of influence spread, has often been approached with a one-size-fits-all mindset. IMSC, in contrast, tailors its seed node selection based on the specific characteristics of each community [14]. This strategic approach, facilitated by the Monarch Butterfly Optimization Algorithm, not only enhances the precision of influence maximization but also marks a departure from rigid methodologies.

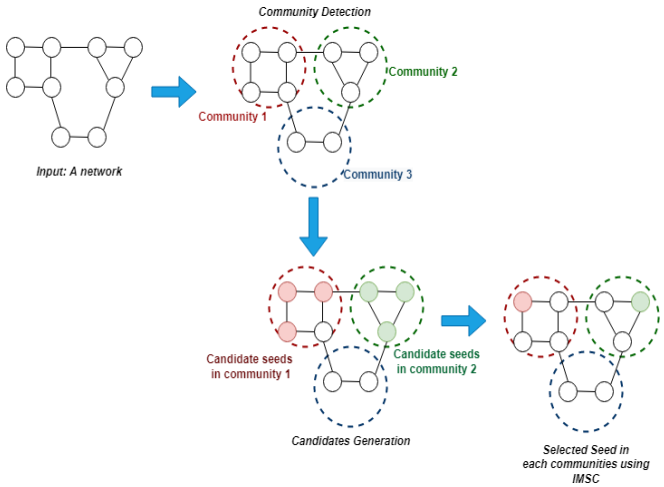


Figure 1. Architecture diagram for the proposed algorithm

To notice the problem of the IM problem, focusing on the diffusion of influence, we offer a novel strategy in this study called IMSC. The three phases of our IMSC approach—candidate generation, community detection, and seed selection—are shown in an overview in Figure 1. In the IMSC, various problems occur. As indicated, community structure offers a way to identify nodes with duplicate influence spreads, eliminating the need for additional influence spread computation. Contribution is,

- We provide a unique approach that uses the network's community spread and seeding phase to maximize information dissemination.
- The experimental findings on four real-world datasets of varying sizes and applications show that the suggested approach performs better than many other IM algorithms.
- We provide a topic-aware, community-based influence maximization method that is superior in efficiency and spread.
- We carry out testing with real datasets. The testing findings show that the IMSC algorithm performs much better than other methods.

The structure of this paper follows as. The study problem is described in Section 2, along with our observations and a review of some pertinent literature. The IMSC method and related algorithms are described in Section 3. The trials on three real datasets are presented in Section 4. Section 5 concludes.

2. PRELIMINARIES

We introduce the basic influence calculation principles in this section, which are necessary for a good understanding of the research.

Notations and definitions

The following notes will be necessary for issue formulations in the study and specific terminologies connected to our proposed study.

Definition 1 (Social network). Graph G represents a social network with M social ties and N users are called a weighted-directed G(V, E, W). The term "influence graph" also refers to a social network.

Definition 2 (Neighbors). The group of users v that form the neighbors N(u) of the u node are described as v ∈ N(u).

Definition 3 (Degree centrality). Degree centrality (CD(u) = |N(u)|) is described as the amount of linkages incident upon a node. We consider node degree in the IM-SSO problem as CD(u) = |Nout(u)|.

Definition 4 (seed nodes). In the social network |S| = k, S ∈ V, the S is a group of nodes that serve as the origin of the data transmission operation.

Definition 5 (Active node). According to the diffusion model, a node u ∈ V, a previously active node, is considered active. Va← {VA ∪ u} after u has been activated.

Definition 6 (Influence spread). influential people the number of users who are still active following the influence process in an influence model, i.e., IS(S) = |VA(S)|.

2.1 Research problem

Their edge represents the social link between two nodes, while a node represents a person (e.g., co-authorship or friendship). A node is designated as present if it has embraced an innovation or an idea or inactive if it has not. As a result, this is the influence maximization dilemma.

Community detection. In a graph, all communities are subgroups of nodes connecting higher edges and small edges separating them. Given the whole graph G = (V, E), divide the vertex set into k subsets, $S_1, S_2, \ldots, S_K$, such that $\cap_{i=1}^{k} S_i = \emptyset$ and$\cap_{i=1}^{k} S_i = V$. A quality metric $Q(S_1, \ldots, S_K)$ is defined over the partitions. This is for non-overlapping community recognition, and to obtain the overlapping version, one need only eliminate the requirement $\cap_{i=1}^{k} S_i = \emptyset$.

Influence maximization problem. The objective is to select k seed nodes that might diffuse their influence on those other seed nodes to increase the number of influenced nodes.

2.2 Related works

We categorize these studies into four groups: (1) Diffusion model, (2) Influence maximization algorithms, (3) Seed user selection in social networks, and (4) community detection-based algorithms.

2.2.1 Diffusion models

The underlying diffusion models, essential to the study of influence maximization, cannot be divided. These diffusion models connect a node with either an active or an inactive state. Active denotes that the node has been persuaded to adopt a new idea or product [15]. The network's nodes are initially entirely dormant. K seed nodes are then enabled as diffusion providers, and influence begins to spread from them until the diffusion process eventually reaches its peak.

IC model: At step t, it will attempt to activate each vertex's presumed inactive neighbors, v, and achieve with probability p_ut for each activated vertex u. When v is effectively activated, it will continue to be active for the remainder of the period, and starting with step t+1, it will similarly influence its neighbors. If v has several active neighbors, each attempt will be made independently. When no more people can be activated, the propagation ends.

LT model: According to the LT model, a person's ability to be activated depends upon the active neighbors in the current step. For each v, the weight parameters must abide by the restriction$\sum_{uneighborofv} b_{uv} \leq 1$. In this configuration, each node v receives an activation threshold θv that is ordinarily chosen randomly from the range [0,1] at the start of diffusion. When no more individuals can be activated, the propagation ceases.

2.2.2 Influence maximization algorithms

The traditional data mining problem of influence maximizing consists of two primary components: the influence maximization method and the propagation model. The algorithm is in charge of identifying a collection of neutron nodes that satisfy the criteria and can maximize the influence of spread, while the propagation model is in charge of extracting and replicating the transfer of information and activation of nodes in the actual network. The related research on increasing impact has been successful as of late. These are the specific introductions.

The authors [16] suggested the Two-stage Iterative structure for the Maximization of Influence in Social Networks (also known as TIFIM). An iterative platform with decreasing order is suggested to choose the nodes of the candidate in the first stage to restrict less significant nodes and lessen the computing difficulty of TIFIM. In particular, the First-Last Allocating Strategy (FLAS) is introduced to calculate the diffuse advantages of every node depending on the outcomes of the previous iteration's two-hop measure. They demonstrate that TIFIM reaches a stable structure within a finite number of rounds. The second stage involves defining ascendant power to assess the phenomenon of diffuse advantage with nodes.

In the study [17], they proposed maximizing the impact's transmission, suggesting a two-level strategy based on the Suspected Infected (SI) epidemic model. They also suggest that using a multithreading strategy to develop the algorithm for the presented SI model will help to improve the algorithm's performance in terms of influence spread per second.

The authors [18] proposed a local influence assessment function to optimize the IM issue in this study. The local influence evaluation function provides a trustworthy anticipated influence spread's diffusion value under the linear threshold, weighted cascade, and independent models. A Learning Automata Based Discrete Particle Swarm Optimization (LAPSOIM) approach is suggested to optimize the local influence. LAPSO-IM reinvents the update rule for particle velocity to address premature convergence about learning automata action.

2.2.3 Seed user's selection in social networks

First, in social network marketing, choosing seed users to optimize their influence is challenging. The phrase mainly refers to identifying a small group of users to increase their influence through the word-of-mouth effect. The information will reach the entire network if these users are made active. Otherwise, the propagation of the knowledge will be stopped. Therefore, the secret is to choose powerful speakers more likely to disseminate the knowledge widely.

In the study [19], the Dynamic Entropy for Influence Maximization (DEIM) algorithm was introduced, whose objective is to recognize the social network's most influential nodes. To find overlapping communities in networks, the Community Overlap Propagation Algorithm based on Cohesive Entropy (CECOPA) is first proposed, and potential nodes in the collecting region are chosen to create the candidate seed set. Then, the ODP method is created using a smaller seed selection range. The introduced DEIM algorithm in this research can effectively harm the optimum amount of users in many settings, as demonstrated by several experiments on various data sets.

A targeted influence maximization issue was put out [20] that considers the seeds' diversity under side information accessible at the values for categorical features at the node level. They created a class of submodular functions and no decreasing monotone to establish the diversity of the category profiles connected to the seed nodes. Two IM methods—one taking advantage of topology-driven diversity and the other considering numerical-based variety in IM—were compared to the researchers' newly created Reverse Influence Sampling (RIS)-based Attribute-Based Diversity-Sensitive Targeted Influence Maximization (ADITUM) algorithm. While acting differently and more adaptable than its rivals, ADITUM gains the advantages of guaranteeing the computational complexity and RIS-typical theoretical guarantee in a broad, categorical-based node diversity context.

2.2.4 Community detection

Some community-based algorithms have been developed to utilize community structure better and reduce the time required to solve the IM problem. Some research projects have recently been proposed utilizing community data to address the influence maximization problem.

To increase performance, the marketing introduced [21] a community-based influence-maximizing approach that incorporates community identification into influence-spreading modeling as opposed to executing it separately. They first construct a thorough latent variable model that accounts for each user's distribution across the community regarding membership, item-topic relevance, and subject interest. Then, they suggest using a collapsed Gibbs sampling approach to train the model. Using community subject interest and topic-irrelevant influence, they then extrapolate community-to-community influence power.

Candidate community formation, community detection, and

seed node selection are the three steps of the method [22]. To be more precise, they first suggest the potential community formation procedure, which uses knowledge of the community framework and node attribute to whittle down the pool of potential community candidates. They then put forth a method to forecast influence power between nodes in a featured network that uses social interaction strength, topology attributes, and structure similarity between nodes to significantly increase prediction accuracy compared to the current approaches. Finally, they suggest the calculation function of influence set to choose seed nodes, allowing the algorithm to be more effective by eliminating 10,000 Monte Carlo models and directly calculating the marginal influence gain of each node.

ComBIM, a community-based method, was developed for resolving the Budgeted Influence Maximization (BIM) issue [23]. The proposed methodology consists of four steps: Community detection to comprehend the social network's inherent structure, budget spreading to distribute the overall spending with the communities based on the number of nodes and their selection costs, seed selection to maximize influence, and budget transfer to transfer any unused budget from one community to another. They use three social network datasets that are openly accessible to implement the suggested methodology. Alternative community detection techniques have also been researched, considering edge content, clique definition, and parallel algorithm. Survey results provide more information.

3. THE FRAMEWORK FOR COMMUNITY-BASED INFLUENCE MAXIMIZING

In this part, we outline the IMSC algorithm and discuss our methods for dealing with the problems it raises. The Monarch Butterfly Optimization Algorithm in IMSC enhances the process of candidate generation. Traditional methods have limitations in efficiently selecting potential influencers within communities. IMSC's optimization ensures a more thorough and practical network exploration, identifying candidates with a higher likelihood of influence. IMSC goes beyond conventional approaches in selecting seed nodes by leveraging the Monarch Butterfly Optimization Algorithm. This results in a more strategic identification of key influencers. Unlike some existing methods that might struggle to adapt to dynamic community changes, IMSC, powered by the Monarch Butterfly Optimization Algorithm, demonstrates high adaptability. It can efficiently adjust its influence maximization strategy as communities evolve, ensuring continued effectiveness. Three phases make up IMSC, as seen earlier in Figure 1: (i) recognition of community, (ii) generating the candidate, and (iii) selecting the seed. Each stage of IMSC is described in the section that follows.

3.1 Community detection in IMSC

A community in a social network is a portion of users who communicate with one another more frequently than users out of the community. We see some potential benefits of investigating community structures in the influence maximization problem, so our first aim is to create a powerful clustering technique for IMSC. An immediate challenge is that such an algorithm needs to achieve our long-term objective of lowering computing costs in the impact maximization problem. The ideas and values of the community may effectively grab human nature activity in social networks. Thus, without utilizing heuristics such as the number of regions, our community detection method seeks to identify the social networks' most organic communities. Be aware that even though an influence maximization task would want to choose four seeds, we shouldn't push a social network to split into four communities if it can accommodate three communities spontaneously. Therefore, this paper aims to build a clustering approach without defining the number of communities to uncover community structures.

Definition. (Influence maximization based on community) Finding a subset S* of k nodes such that the objective method specified in Eq. (1) is maximized is the goal of the influence maximization based on community issue given a network G(V; E) with a non-overlapping community structure C. i.e.,

$$S^* = arg_S \max f(S) \tag{1}$$

Let $P_v(S, S', C)$ represent the likelihood that node v in community C will eventually become active. If so, the computation of $f(S, S', C)$is as follows:

$$f(S, S', C) = \sum_{v \in C} P_v(S, S', C) \tag{2}$$

Although Eq. (2)'s form is unique, the computation of $P_v(S, S', C)$ is open. It is simple to observe that any $v \in S$, we have $P_v(S, S', C) = 1$. For any $v \in S'$, i.e., $v \in N(S)$, we have $P_v(S, S', C) = 1 - \Pi_{u \in N(v) \cap S(1-p_{uv})}$. The calculation of the key value is $P_v(S, S', C)$ for the rest vertices. The method figures out the different groups or communities within a larger network. Imagine a social media platform - it's like identifying distinct groups of friends who frequently interact with each other.

3.2 Candidate generation

The candidate generation section seeks to identify a collection of candidate seeds depending on the number of groups and the connection of the nodes across groups in light of the found community framework. The search space for choosing seeds with the most significant influence diffuse is enormous due to the size of social networks in realistic settings. As a result, it's essential to lower the number of candidate seeds efficiently. One of the main problems IMSC is facing at this point is how to reduce the quantity of the candidate seeds.

According to our observations, the seeds chosen from a broad community may lead to more people accepting a thing or an invention than seeds chosen from a little group. As a result, choosing the center nodes of the social network's k-largest groups as the k-influential seeds is an intuitive strategy. However, this naive approach has a few potential issues: (1) The finding above suggests that in significant communities, we should choose more seeds, and (2) Some critical data about the community framework is disregarded. Be aware that the introduced MBO not only identifies hubs but also communities. While community centroids may seem apparent candidates for seed selection, the portal should also be considered because they can quickly extend their effects throughout numerous communities.

Definition 3 (Significant Communities)

MBO produces a collection of communities, indicated as $C = \{c_1, c_2, \ldots\ldots, c_p\}$. The following is a definition of the set

of significant community C_s:

$$C_s = \left\{ c_j \in C | n_j \geq \frac{\sum_{i=1}^{p} n_i}{k} \right\} \quad (3)$$

Once the communities are identified, the method looks for potential candidates to be influential or essential within these groups. Think of it as finding individuals who could significantly impact their respective friend circles or communities.

3.3 Seed selection

The community detection separates the social network into many communities, and each community's gain is then assessed using node coverage gain. We choose suitable nodes as seeds by using the outcomes of the node coverage gain evaluation and community detection. Seed nodes will be chosen from all the other communities, other than that, from only a few supposedly important groups, to minimize the rich-club effect and increase the impact dissemination. The following is a description of the three-step seed-choosing strategy:

Step 1: There is a largest-gain node inside each community. The node with the highest gain will be given preference when choosing a fresh seed if the node's gain values differ.

Step 2: The community node with no more seeds will be chosen as a new seed if more than one node in the entire network in Step 1 has the same maximum benefit.

Step 3: If many nodes in the network have the same maximum gain and every associated community has a minimum of one seed in Step 2, then the community node with the most extensive scale will be chosen as a new seed.

The most influential node, the seed, should be chosen for the initial step if a node has the highest gain value. We should prioritize the large-scale community; the second step has no seeds, and the third does not. It should be emphasized that just one seed node is chosen each time using this three-step method. The coverage gain of the available ones will be maintained for the subsequent chosen when a seed node has been chosen.
Now, from the pool of potential candidates, the method selects the best starting points, or "seed nodes." These seed nodes are like critical community influencers, serving as initial points for spreading information or influence.

3.4 Monarch Butterfly Optimization (MBO)

Swarm intelligence algorithms are believed to include the population-based algorithm known as the MBO. The behavior of certain insects, such as bees and butterflies, is what inspired these algorithms. As stated earlier, the MBO was developed in the modern era. The exquisite form of a butterfly species exclusive to North America and recognized by its orange and black colors served as the model for this ingenious program [24]. Twice a year, the monarch butterfly migrates, like many other butterflies. The first relocation originates in Canada and travels south to Mexico, while the second is upward migration from Mexico to Canada. Simulating these butterflies' migratory behavior solves several optimization problems. A few rules and fundamental concepts need to be adhered to get the optimal solution to the problem:

1. All butterflies are made of people who live in either Land 1 or Land 2 (the home following migration).

2. The migration operator produces Every butterfly's progeny, regardless of whether the parents are land 1 or inland 2.

3. A candidate function will remove one of the two since the population shouldn't fluctuate and should always remain constant.

4. The migration operator leaves the butterflies selected by the candidate function intact, passing them on to the next generation.

The butterflies begin their annual migration in early April when they depart from land l and travel to land 2. The return journey starts in September. An overall count of monarch butterflies in the two countries accurately represents the entire population or NP.

3.4.1 Migration facilitator

The following is a representation of the butterfly relocation process:

$$X_{i.k}^{t+1} = X_{r1.k}^{t} \quad (4)$$

The location of a butterfly, *i*, is indicated by $X_{r1.k}^{t}$, which represents the Kth components of the freshly generated location, and $X_{i.k}^{t+1}$, which represents the Kth components of Xi at the t + 1 generation. In this instance, the following equation was used to create the random integer *r*:

$$R = rand * peri \quad (5)$$

where, *peri* denotes the period of the migration.

However, if $r > p$, the Kth variables determining the location of the next generation are found using the equation that follows:

$$X_{i.k}^{t+1} = X_{r2.k}^{t} \quad (6)$$

where, $X_{r2.k}^{t}$ represents the Kth generation of X_{r2}'s constituents for the butterfly r2. As a result, *P* represents the quantity of monarch butterflies present per acre.

3.4.2 Butterfly adjusting operator

To acquire the locations of the monarch butterflies, in addition to the migration operator, the following butterfly adjustment operator can also be used [25]. The butterfly adjusting operator method can be summed up in the following words. An erratically generated number can be improved if it is less than or equal to *p* for each of the monarch butterfly *j* constituent parts.

$$X_{j.k}^{t+1} = X_{best.k}^{t} \quad (7)$$

$X_{j.k}^{t+1}$is the kth component of X_j at generation t + 1, which displays the monarch butterfly's j position.

$$X_{j.k}^{t+1} = X_{r3.k}^{t} \quad (8)$$

Xr3.kt denotes the randomly selected kth element from x r3 in Land 2. This instance has $r3 \in \{1,2,\ldots,NP_2\}$. In this case, it can be updated further as indicated below if rand>BAR.

$$X_{j.k}^{t+1} = X_{j.k}^{t+1} + \alpha \times (dx_k - 0.5) \quad (9)$$

where, BAR is the butterfly's adjustment rate, a monarch butterfly's walk step or dx can be ascertained via Levy flying.

$$dx = Levy\left(x_j^t\right) \quad (10)$$

The weighting factor in Eq. (9), denoted by α, is derived from Eq. (11).

$$\alpha = S_{max}/t^2 \quad (11)$$

While the small, representing a quick step in the search process, lowers the influence of dx on $x_{j.k}^{t+1}$ and encourages the exploitation phase, the bigger, representing a longer step in the search process, enhances the influence of dx on $x_{j.k}^{t+1}$. Consequently, it can characterize the individual who modifies the butterfly.

3.5 IMSC algorithm

The algorithm for the entire IMSC framework is based on the above captions for every phase. Monarch Butterfly Optimization can get data about the hubs and community frameworks—an overview of the IMSC algorithm.

All the parameters are set up before creating the initial population and assessing it using its fitness function. The whereabouts of each monarch butterfly are then gradually updated until specific standards are fulfilled. It should be noted that the quantity of monarch butterflies produced by the butterfly adjustment operator and migratory operator, respectively, is NP2 and NP1, to fix the population and community selections.

Algorithm 1. IMSC Algorithm.
Begin

Step 1: Initialization. Set the generation counter to 1 and randomly choose individuals from the population P of the community; Set the maximum generation MaxGen, the number of communities Land 1 and NP_1 in Land 1, the maximum step S_{max}, the community adjustment pace BAR, the migration ratio and the migration period peri.

Step 2: Community detection. Consider each community in light of its location.

Step 3: While the ideal answer has not been discovered**or** t<MaxGen**do**

Sort each candidate based on how connected they are.

Separate the community of peoples into the Land 2 and Land 1 subpopulations;

For i=1to NP_1 (for all communities in Subpopulation 1) **do**

Generate new Subpopulation 1.

end for j

for j=1 to NP_2 (for all communities in Subpopulation 2) **do**

Generate new Subpopulations 2.

end for j

Then, select the seed node from the generated candidates.

Evaluate the populace in light of the most recent adjustments t=t+1.

Step 4: end while

Step 5: Produce the ideal answer.

End

MBOA can adapt to changing community dynamics. It's like having a strategy that can adjust and evolve as the social structure of a community changes over time, ensuring continued effectiveness. MBOA excels in efficiently exploring the network to find influential nodes. It's like having a team of efficient scouts that systematically search and identify the most impactful individuals within a community. Communities can have diverse structures, and MBOA can optimize for various types. Whether a community is tightly knit or loosely connected, MBOA can identify and maximize influence across different structures, making it versatile. MBOA's optimization process converges to optimal solutions relatively quickly. This means it efficiently identifies the most influential nodes without unnecessary delays, making it a time-effective approach.

4. EVALUATION

We examined the algorithms' running times, the caliber of the seed nodes, and memory use on four real datasets to see how well our approach compares to alternative approaches.

4.1 Real-world networks

We first assess how well our community-based impact maximization approach performs on four real-world datasets that range in size from small to big and represent the volume and diversity aspects of big data [26]. The most enormous dataset has roughly 117 million edges and 3.1 million nodes. To assess the performance and demonstrate the application of the IMSC algorithm, we also conducted several experiments on four real datasets: (1) NETHept, (2) the Amazon, (3) the Epinions, and (4) the DBLP.

4.1.1 NetHEPT

The electronic publishing platform arXiv (http://www.arxiv.org/) offers the NetHEPT dataset. The network has ties between writers who collaborate in high energy physics theory. An undirected edge will form between authors j and I if they co-author at least one paper. There are 31K edges and 15K nodes in the dataset.

4.1.2 Amazon

The Stanford Large Network Dataset Collection (SNAP), a free online library made available, is where the Amazon dataset may be found. They indexed the data from Amazon's online store (http://www.amazon.com). An undirected edge between product I and product j is formed if they are regularly purchased together. There are 335K nodes and 926K edges in the network.

4.1.3 Epinions

The social network Epinions is part of the customer review website Epinions.com. Customers will have an advantage over one another if they believe the reviews of other customers. After combining all repeated edges, the network has 406K edges and 76K vertices.

4.1.4 DBLP

Another dataset obtained through the DBLP platform is DBLP, which enables co-authorship between academics in computer science [27]. Each edge in the network, which has 317K vertices and 1M edges, shows that two researchers to whose vertices it corresponds have co-authored at least one

paper.

4.2 Algorithmic comparison

INSC: The algorithm introduced in this research.

LDAG: We used the authors' suggested parameter setting = 1/320.

SIMPATH: The authors' suggested values for the algorithm are l = 4 and = 103. The Table 1 illustrates the community information for each dataset.

Table 1. Community information for the datasets

	NetHEPT	Amazon	Epinions	DBLP
#Communities	2,262	12,326	10,307	2,520
#Nodes in the most significant community	15K	335K	76K	317K
#Edges in the most significant community	31K	926K	406K	1M

IPA: The author runs the suggested algorithm with a threshold of 0.005.

PageRank: In this study, the seed nodes have the highest ranking. When the scoring vector from two successive iterations following one another is different by no more than 106 as measured by the L1 norm, the algorithm ends.

HighDegree: This technique selects the highest out-degree nodes as seed nodes.

We contrast IMSC with the algorithms above for the following reasons: Two known well and fundamental approaches combined with most other works are HighDegree and PageRank. SAMPATH is a method that performs well in terms of execution time, memory utilization, and seed node quality. In terms of seed node quality and response times, LDAG also performs well. Finally, IPA looks for the most influential nodes using the community concept.

4.3 Experimental results

Analyzing memory utilization because they don't want to keep any frameworks while operating, the HighDegree and PageRank algorithms require almost minimal memory. In contrast, the LDAG method uses the most significant memory because it creates a DAG for every node in the graph. IMSC requires less memory than LDAG but more than SIMPATH. Figure 2 displays the outcomes for 50 seed sets. The y-axis in this image has a base-10 logarithmic scale. The research contributes significantly by introducing a novel method, IMSC that effectively identifies influential individuals within communities. It adds a valuable layer to understanding social networks and community dynamics. Think of it as discovering a new and more efficient way to pinpoint critical players in a social group.

Evaluating seed set quality, the algorithm that spreads its impact more widely is one of the better qualities. According to the experiment, the IMSC algorithm has the best seed set out of all the algorithms. Even though IMSC employs the SIMPATH approach to compute the widespread, its diffuse set of seeds is generally a tiny better than SIMPATH because IMSC combines local and global spreads, and so keeps account of both each node's impact on its community and the impact of each community as a whole.

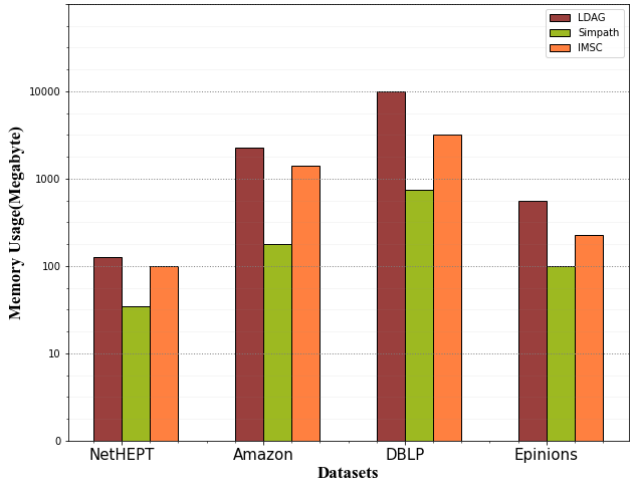


Figure 2. Comparison of various algorithms' memory use

In every dataset but the NetHept dataset, the IPA algorithm's seed sets are poorer than the other techniques. The IPA algorithm's seed set quality is the lowest in netHept and Amazon, while in DBLP, it is more significant than HighDegreeand PageRank but lower than IMSC, LDAG, and SIMPATH. The IPA algorithm could be more dependable because, for some other datasets, its seed nodes are better than other algorithms in terms of quality. Still, for the majority of others, it leads to worse quality.

Analyzing running times, the outcomes based on running times comparisons are displayed in Figure 3. The charts for PageRank and HighDegree in the NetHept dataset are removed because of the low running times of these algorithms. IPA runs faster than IMSC, SIMPATh, and LDAG and is comparable in speed to PageRank and HighDegree. Besides IPA, HighDegree, and PageRank, the IMSC algorithm has the fastest running time.

Figures 3(a) and 3(d) show that IMSC performs less well than SIMPATH early in its running time. For example, in Figure 3a, the time required to discover seed nodes up to 15 is longer than the time required by the SIMPATH approach.

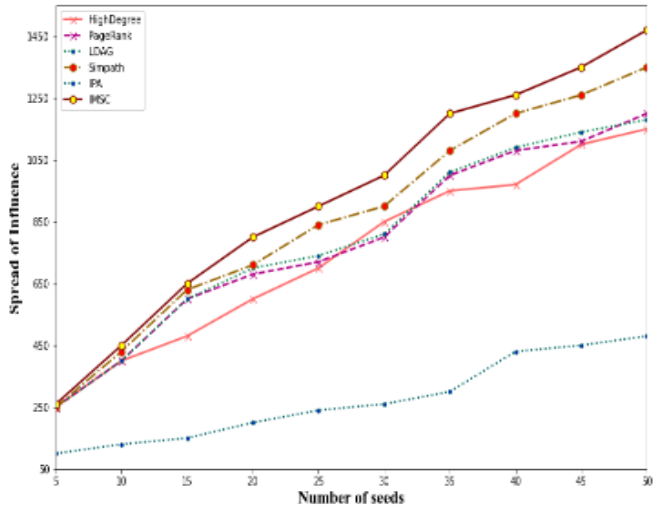
The communities that the algorithm takes into account need to be changed. With 50 seed nodes to choose from, Table 2 contrasts the running times of the suggested algorithm with those of LDAG and SIMPATH, which have acceptable running times and a higher set of seed performance. As we can see, depending on the dataset, our strategy improves running time by 27 and 86 percent.

Every time the algorithm iterates, each node diffuse is calculated locally in its neighborhood. Additionally, the diffuse of the other nodes in community Ci, as well as the nodes spread in communities with a direct way from Ci, should be modified when node v is selected from community Ci. As a result, rather than modifying all of the nodes in the graph, the spread of a select few would be updated in each cycle. This contribution makes subsequent iterations of the algorithm run more quickly.

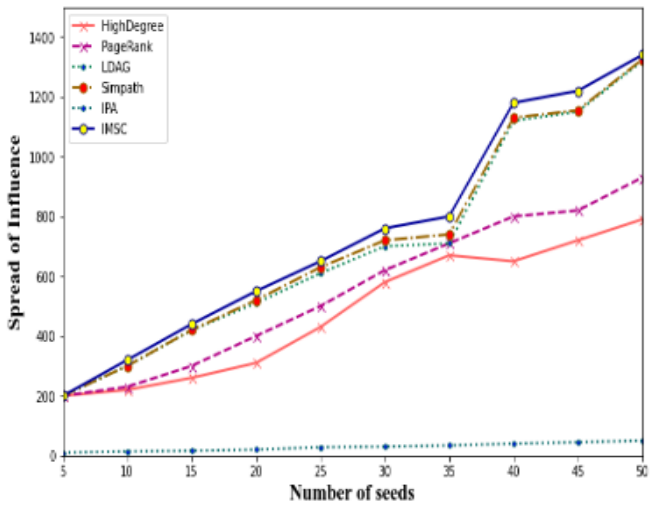
To determine the ultimate spread computation's effectiveness, we compare the spread obtained by IMSC and the simple greedy algorithm to determine how well our method calculates the distribution of nodes. To calculate the diffusion of the seed sets, we conduct IMSC and MC models10,000 times using five distinct randomly selected sets of nodes with

sizes 10, 20, 30, 40, and 50 as the seed sets. The outcomes are displayed in Figure 4.

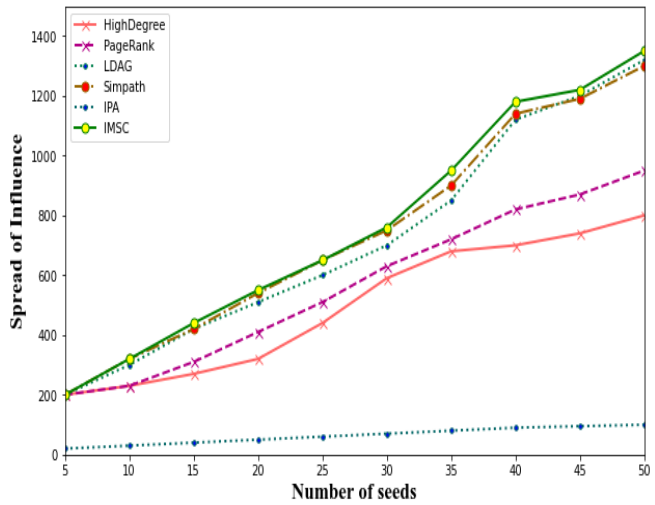
The values calculated by IMSC are remarkably similar to those calculated by MC models for various sets of nodes, as shown in Figure 4. For sets of sizes 10, 20, 30, 40, and 50, respectively, there are 0.68 percent, 0.5 percent, 0.8 percent, 0.62 percent, and 0.85 percent differences in the values computed by IMSC and MC. Additionally, for the DBLP dataset, the differences in values for sizes 10, 20, 30, 40, and 50 are 0.8 percent, 0.69 percent, 0.73 percent, 0.32 percent, and 0.47 percent.



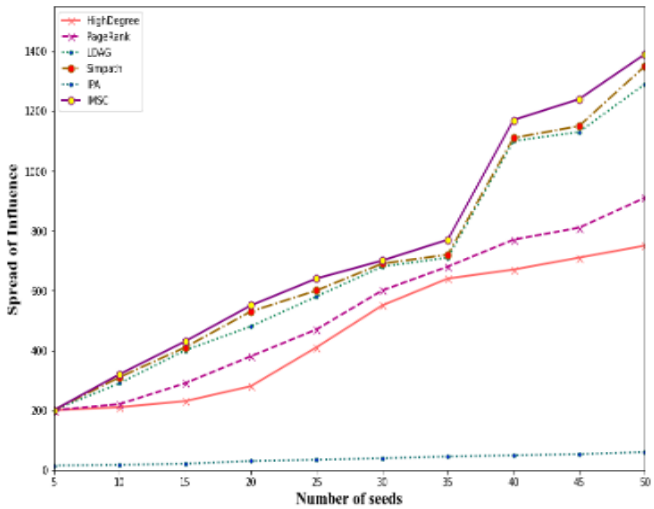
(a) NetHEPT



(b) Amazon

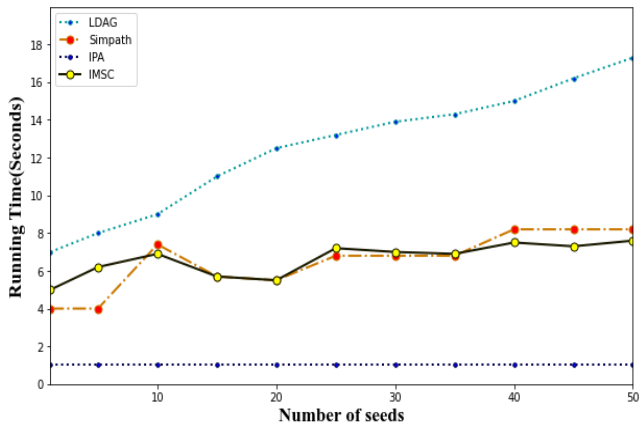


(c) Epinions

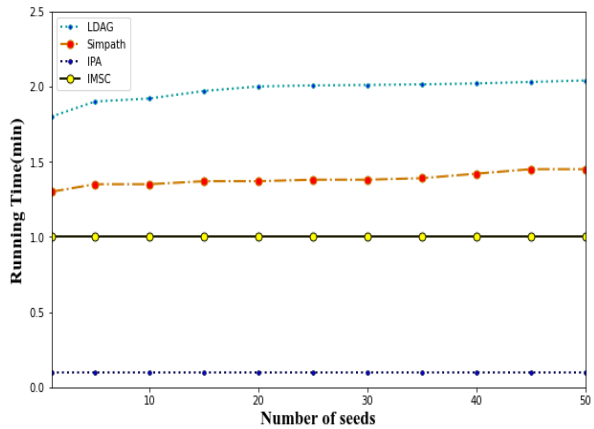


(d) DBLP

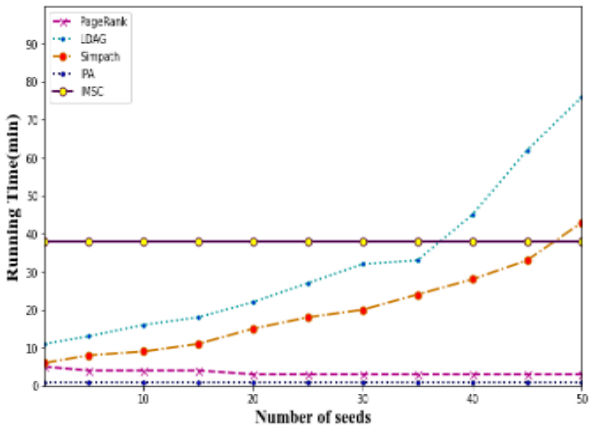
Figure 3. Influence spread made possible by several algorithms



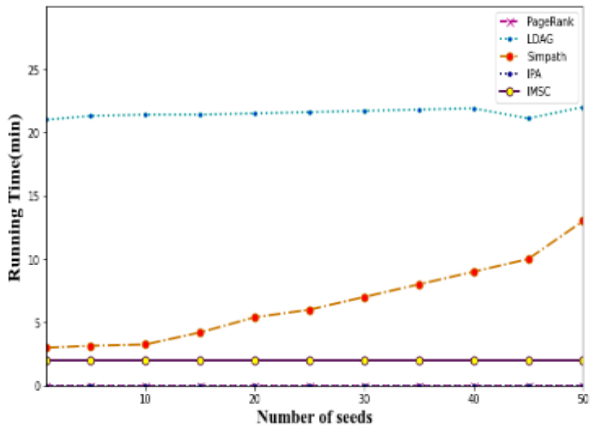
(a) NetHEPT



(b) Amazon



(c) Epinions



(d) DBLP

Figure 4. Comparing the running time of several methods

Table 2. Increase speed for various datasets

Compared Algorithms	Datasets			
	NetHEPT	Amazon	Epinions	DBLP
Simpath	43%	27%	68%	54%
LDAG	54%	53%	86%	67%

These outcomes demonstrate that our method efficiently calculates the diffuse values by fusing every node's global diffuse and local diffuse. The spread given by different types of seed sets selected at random is displayed in Figure 5.

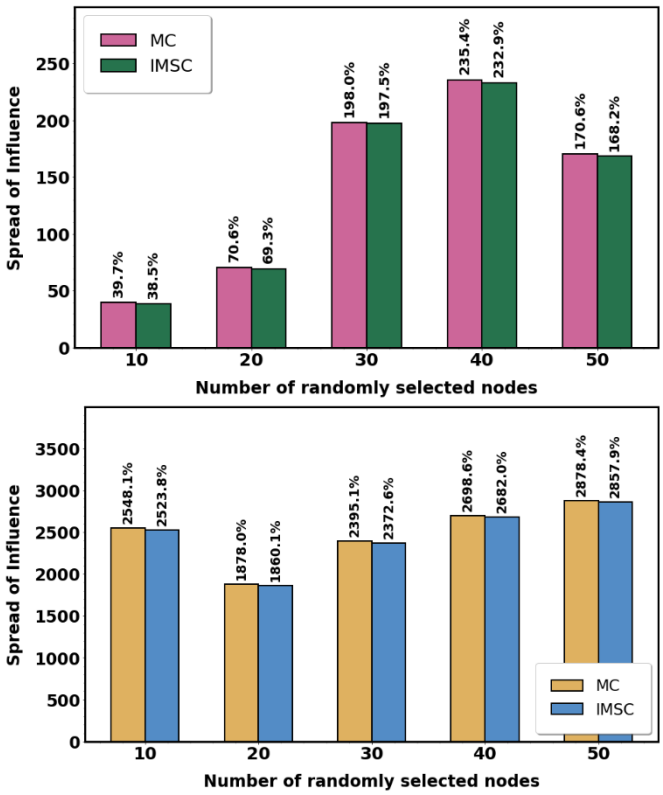


Figure 5. The spread produced by various seed sets chosen at random

The findings of this research have real-world applications, especially in areas where influencing communities is crucial. For example, in marketing, you can use this method to identify the best starting points for spreading a message within specific customer segments. It's like having a smart strategy to ensure your message reaches the most influential people in a community, maximizing its impact. This method can be applied in social initiatives to identify key community leaders or influencers who can drive positive change. It's akin to finding the most effective champions within a group to lead and amplify efforts for a cause.

5. CONCLUSIONS

In this research, the IMPC framework is proposed as a solution to the problem of IM on community networks. This paper aims to notice the influence maximizing and efficiency issues without sacrificing the effectiveness of the influence spread that was attained using the heat diffusion model. Social networks are becoming more extensive, making efficient mining algorithms essential for many applications. To our knowledge, the influence maximization problem has never been solved using community structure before IMSC. Many existing methods for identifying critical influencers in communities often need help need help. They might need help locating influential individuals, leading to less effective strategies. It's like finding essential figures in a crowd without a clear map. The IMSC approach stands out by addressing these challenges head-on. It acts like a more accurate and efficient map, overcoming the limitations of existing methods. Identifying community structures and key players ensures that the selected influencers have a genuine impact. The experimental outcomes on actual and fictitious datasets demonstrate that our suggested IMSC outperforms existing methods regarding running duration and influence spread. Depending on the outcomes, IMSC will be enhanced in further work by using influence to activate various nodes and manage propagation over the entire social media network graph.

REFERENCES

[1] Tang, J., Zhang, R., Wang, P., Zhao, Z., Fan, L., Liu, X. (2020). A discrete shuffled frog-leaping algorithm to identify influential nodes for influence maximization in social networks. Knowledge-Based Systems, 187: 104833. https://doi.org/10.1016/j.knosys.2019.07.004

[2] Keikha, M.M., Rahgozar, M., Asadpour, M., Abdollahi, M.F. (2020). Influence maximization across

heterogeneous interconnected networks based on deep learning. Expert Systems with Applications, 140: 112905. https://doi.org/10.1016/j.eswa.2019.112905

[3] Zareie, A., Sheikhahmadi, A., Jalili, M. (2020). Identification of influential users in social network using gray wolf optimization algorithm. Expert Systems with Applications, 142: 112971. https://doi.org/10.1016/j.eswa.2019.112971

[4] Ju, W., Chen, L., Li, B., Liu, W., Sheng, J., Wang, Y. (2020). A new algorithm for positive influence maximization in signed networks. Information Sciences, 512: 1571-1591. https://doi.org/10.1016/j.ins.2019.10.061

[5] Singh, S.S., Singh, K., Kumar, A., Biswas, B. (2020). ACO-IM: Maximizing influence in social networks using ant colony optimization. Soft Computing, 24(13): 10181-10203. https://doi.org/10.1007/s00500-019-04533-y

[6] Tian, S., Mo, S., Wang, L., Peng, Z. (2020). Deep reinforcement learning-based approach to tackle topic-aware influence maximization. Data Science and Engineering, 5(1): 1-11. https://doi.org/10.1007/s41019-020-00117-1

[7] Raghavan, S., Zhang, R. (2022). Rapid influence maximization on social networks: The positive influence dominating set problem. INFORMS Journal on Computing, 34(3): 1305-1840. https://doi.org/10.1287/ijoc.2021.1144

[8] Shang, J., Zhou, S., Li, X., Liu, L., Wu, H. (2017). CoFIM: A community-based framework for influence maximization on large-scale networks. Knowledge-Based Systems, 117: 88-100. https://doi.org/10.1016/j.knosys.2016.09.029

[9] Li, J., Cai, T., Deng, K., Wang, X., Sellis, T., Xia, F. (2020). Community-diversified influence maximization in social networks. Information Systems, 92: 101522. https://doi.org/10.1016/j.is.2020.101522

[10] Chen, X., Deng, L., Zhao, Y., Zhou, X., Zheng, K. (2021). Community-based influence maximization in location-based social network. World Wide Web, 24(6): 1903-1928. https://doi.org/10.1007/s11280-021-00935-x

[11] Wang, Z., Sun, C., Xi, J., Li, X. (2021). Influence maximization in social graphs based on community structure and node coverage gain. Future Generation Computer Systems, 118: 327-338. https://doi.org/10.1016/j.future.2021.01.025

[12] Bozorgi, A., Haghighi, H., Zahedi, M.S., Rezvani, M. (2016). INCIM: A community-based algorithm for influence maximization problem under the linear threshold model. Information Processing & Management, 52(6): 1188-1199. https://doi.org/10.1016/j.ipm.2016.05.006

[13] Wang, X., Tong, X., Fan, H., Wang, C., Li, J., Wang, X. (2021). Multi-community influence maximization in device-to-device social networks. Knowledge-Based Systems, 221: 106944. https://doi.org/10.1016/j.knosys.2021.106944

[14] Li, X., Cheng, X., Su, S., Sun, C. (2018). Community-based seeds selection algorithm for location aware influence maximization. Neurocomputing, 275: 1601-1613. https://doi.org/10.1016/j.neucom.2017.10.007

[15] Singh, S.S., Kumar, A., Singh, K., Biswas, B. (2019). C2IM: Community based context-aware influence maximization in social networks. Physica A: Statistical Mechanics and Its Applications, 514: 796-818. https://doi.org/10.1016/j.physa.2018.09.142

[16] He, Q., Wang, X., Lei, Z., Huang, M., Cai, Y., Ma, L. (2019). TIFIM: A two-stage iterative framework for influence maximization in social networks. Applied Mathematics and Computation, 354: 338-352. https://doi.org/10.1016/j.amc.2019.02.056

[17] More, J.S., Lingam, C. (2019). A SI model for social media influencer maximization. Applied Computing and Informatics, 15(2): 102-108. https://doi.org/10.1016/j.aci.2017.11.001

[18] Singh, S.S., Kumar, A., Singh, K., Biswas, B. (2019). LAPSO-IM: A learning-based influence maximization approach for social networks. Applied Soft Computing, 82: 105554. https://doi.org/10.1016/j.asoc.2019.105554

[19] Li, W., Zhong, K., Wang, J., Chen, D. (2021). A dynamic algorithm based on cohesive entropy for influence maximization in social networks. Expert Systems with Applications, 169: 114207. https://doi.org/10.1016/j.eswa.2020.114207

[20] Caliò, A., Tagarelli, A. (2021). Attribute based diversification of seeds for targeted influence maximization. Information Sciences, 546: 1273-1305. https://doi.org/10.1016/j.ins.2020.08.093

[21] Huang, H., Shen, H., Meng, Z., Chang, H., He, H. (2019). Community-based influence maximization for viral marketing. Applied Intelligence, 49(6): 2137-2150. https://doi.org/10.1007/s10489-018-1387-8

[22] Huang, H., Shen, H., Meng, Z. (2020). Community-based influence maximization in attributed networks. Applied Intelligence, 50(2): 354-364. https://doi.org/10.1007/s10489-019-01529-x

[23] Banerjee, S., Jenamani, M., Pratihar, D.K. (2019). ComBIM: A community-based solution approach for the budgeted influence maximization problem. Expert Systems with Applications, 125: 1-13. https://doi.org/10.1016/j.eswa.2019.01.070

[24] Bozorgi, A., Samet, S., Kwisthout, J., Wareham, T. (2017). Community-based influence maximization in social networks under a competitive linear threshold model. Knowledge-Based Systems, 134: 149-158. https://doi.org/10.1016/j.knosys.2017.07.029

[25] Kumar, S., Singhla, L., Jindal, K., Grover, K., Panda, B.S. (2021). IM-ELPR: Influence maximization in social networks using label propagation based community structure. Applied Intelligence, 51(11): 7647-7665. https://doi.org/10.1007/s10489-021-02266-w

[26] Samir, A.M., Rady, S., Gharib, T.F. (2021). LKG: A fast scalable community-based approach for influence maximization problem in social networks. Physica A: Statistical Mechanics and Its Applications, 582: 126258. https://doi.org/10.1016/j.physa.2021.126258

[27] Beni, H.A., Bouyer, A. (2020). TI-SC: Top-k influential nodes selection based on community detection and scoring criteria in social networks. Journal of Ambient Intelligence and Humanized Computing, 11(11): 4889-4908. https://doi.org/10.1007/s12652-020-01760-2